Docket No. AUS9-2000-0728-US1

CLAIMS:

What is claimed is:

1. A method for generating pseudo random test patterns for simulating a hardware model comprising:

generating a driver model having a plurality of states, wherein each state indicates whether to drive an interface of the hardware model;

initiating a random walk through the driver model to generate a driver test pattern; and

controlling simulation of the hardware model using the driver test pattern.

2. The method of claim 1, wherein each state of the plurality of states comprises one of a drive state and a wait state.

3. The method of claim 1, wherein the step of generating a driver model comprises:

creating at least one driver subgraph having a plurality of states; and

connecting the at least one driver subgraph to form the driver model.

4. The method of claim 3, wherein each driver subgraph is a Markov chain.

5. The method of claim 3, wherein each state has a probability of transitioning to at least one other state.

6. The method of claim 1, further comprising:

generating a command model having a plurality of states, wherein each state indicates a command to send across an interface of the hardware model;

initiating a random walk through the command model to generate a command test pattern; and

controlling simulation of the hardware model using the command test pattern.

7. The method of claim 6, wherein the step of generating a command model comprises:

creating at least one command subgraph having a plurality of command states; and

connecting the at least one command subgraph to form the command model.

8. The method of claim 7, wherein each command subgraph comprises a Markov chain.

9. The method of claim 7, wherein each state has a probability of transitioning to at least one other state.

10. An apparatus for generating pseudo random test patterns for simulating a hardware model comprising:

generation means for generating a driver model having a plurality of states, wherein each state indicates whether to drive an interface of the hardware model;

initiation means for initiating a random walk through the driver model to generate a driver test pattern; and

control means for controlling simulation of the hardware model using the driver test pattern.

11. The apparatus of claim 10, wherein each state of the plurality of states comprises one of a drive state and a wait state.

12. The apparatus of claim 10, wherein the generation means comprises:

means for creating at least one driver subgraph having a plurality of states; and

means for connecting the at least one driver subgraph to form the driver model.

13. The apparatus of claim 12, wherein each driver subgraph is a Markov chain.

14. The apparatus of claim 12, wherein each state has a probability of transitioning to at least one other state.

15. The apparatus of claim 10, further comprising:

means for generating a command model having a plurality of states, wherein each state indicates a command to send across an interface of the hardware model;

means for initiating a random walk through the command model to generate a command test pattern; and

means for controlling simulation of the hardware model using the command test pattern.

16. The apparatus of claim 15, wherein the means for

generating a command model comprises:

means for creating at least one command subgraph having a plurality of command states; and

means for connecting the at least one command subgraph to form the command model.

17. The apparatus of claim 16, wherein each command subgraph comprises a Markov chain.

18. The apparatus of claim 16, wherein each state has a probability of transitioning to at least one other state.

19. A computer program product, in a computer readable medium, for generating pseudo random test patterns for simulating a hardware model comprising:

instructions for generating a driver model having a plurality of states, wherein each state indicates whether to drive an interface of the hardware model;

instructions for initiating a random walk through the driver model to generate a driver test pattern; and

instructions for controlling simulation of the hardware model using the driver test pattern.

20. The computer program product of claim 19, further comprising:

instructions for generating a command model having a plurality of states, wherein each state indicates a command to send across an interface of the hardware model;

instructions for initiating a random walk through the command model to generate a command test pattern; and

instructions for controlling simulation of the hardware model using the command test pattern.